2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Contacts from Motion: Learning Discrete Features for Automatic Contact Detection and Estimation from Human Movements*

Hibiki Miyake[1], Ko Ayusawa[2,3], Ryusuke Sagawa[3,4] and Eiichi Yoshida[1]

***Abstract*— This paper presents a novel method for detecting and estimating contact forces only from human motions using machine learning techniques. Knowing the location of the contacts with the environment and the magnitude of the exerted force is critical for dynamic human motion analysis. However, their annotation is usually made manually from captured motion data especially in case of multiple contacts even if the data includes force measurement. Moreover, most existing human motion datasets do not include contact force. To overcome these bottlenecks, we introduce a network that leverages vector-quantized variational autoencoder (VQ-VAE) and self-attention that learns a small set of discrete feature values representing various contact states. These feature values, called contact codes, allow human motions to be converted to contact states and resulting forces. By applying an optimization for contact estimation with a reduced set of manual annotations, the existence of contacts can be automatically determined, which is essential information for dynamic analysis. We validated the effectiveness and potential usefulness of the proposed method with a human walking gait dataset, by converting the human motions into contact sequences and forces and applying the estimated contacts to dynamic motion analysis.**

I. Introduction

Research on human motion analysis and understanding is becoming more and more important in many related areas such as computer graphics, robotics, biomechanics, and medical applications. In a robotics context, natural robot motion generation and human-robot interaction (HRI) are relevant topics that can benefit from this research. Physical interaction is inevitable for robots to be well-accepted in society. In our daily lives, it appears everywhere involving both environments and other humans: in addition to walking, sitting, moving objects, and using tools as well as human-human interaction such as handover, collaborative tasks like carrying, or direct contacts like handshaking. Robots that can appropriately deal with those physical interactions can accelerate their societal integration.

This leads to one of the important challenges, understanding contacts in human motions [1] that have several complexities, making their modeling and analysis challenging. Measuring contact force needs additional devices, if multiple contacts are involved, then force measurement needs many such sensors. Another difficulty lies in measuring the force precisely, in particular, because contacts usually occur on surfaces, not points as often modeled in robotics. In addition, contact description includes discrete status and continuous values where the former detects whether contacts exist and where they are if any, and the latter quantifies the magnitude and direction of the contact force. Contact detection is essential for dynamics analysis since changes in contact constraints make differences in resultant motions. However, annotation of contacts in the measured human motion data is a cumbersome task, requiring the operator to visually check the contacts with the graphical display of captured human motions.

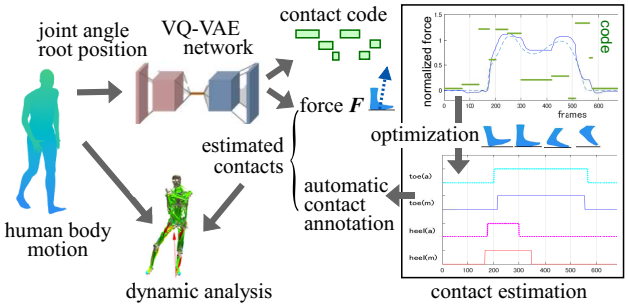

Fig. 1. Overview of contact estimation from motion. After learning the "contact codes" relating motion and contact force, the network becomes capable of converting motions to contact force and state, which can be utilized by dynamic analysis.

Recent advances in computer graphics allow estimating contact spots from 3D scenes [2], [3] to address this issue. While this work is useful for understanding interactions with environments, contact detection accuracy still requires improvements to be exploited in dynamic analysis. Therefore, in order to detect contacts from available human motion databases including contact information [4], [5], manual annotation of contact occurrence is still necessary, especially for contacts that cannot be distinguished even with force information such as "toe-touchdown" and "heel-liftoff" during the support phase of walking. Automatic contact detection from motions could significantly reduce such manual labor to improve the efficiency of human motion analysis to provide contact-rich human motion datasets. Moreover, such automatic contact detection and estimation have the potential to turn contactless human motions into contact-rich data. Nevertheless, a systematic method is yet to be developed to detect and estimate contacts from measured motions.

This paper presents a novel method for both automatic

*This research was mainly supported by JSPS KAKENHI Scientific Research (S) Grant Number 22H05002, and partially supported by JSPS KAKENHI Scientific Research (A) Grant Number 22H00545.

[1]Hibiki Miyake and Eiichi Yoshida are with Department of Medical and Robotic Engineering Design, Faculty of Advanced Engineering, Tokyo University of Science, Tokyo, Japan. 8123547@ed.tus.ac.jp, eiichi.yoshida@rs.tus.ac.jp

[2]Ko Ayusawa is with Human Augmented Research Center (Kashiwa, Chiba) and [3]CNRS-AIST JRL (Joint Robotics Laboratory), IRL, [4]Ryusuke Sagawa is with Artificial Intelligence Research Center (Tsukuba, Ibaraki), National Institute of Advanced Industrial Science and Technology (AIST), Japan. {k.ayusawa,ryusuke.sagawa}@aist.go.jp

contact detection and force estimation from the input motion only, leveraging self-supervising machine learning techniques that combine the mechanism of "Self-attention" [6] and vector-quantized variable autoencoder (VQ-VAE) [7], [8]. The former features its capacity to model a relationship between the time-series input data, whereas the latter network extracts latent force values with reduced dimensions together with discrete feature values that can be utilized to detect the addition and removal of contacts. This method has been applied to the input of whole-body human motion represented by a generalized digital human model "Dhaiba" [9] to learn the output of finite discrete contact state and continuous force.

In this paper, with the example of human walking from AIST Gait Database [5] with ground-truth contact force information, we demonstrate that the discrete feature values, referred to as "contact codes," and reconstructed force value obtained from the trained VQ-VAE network correspond to the actual contact state and force accurately enough. We also show that the output of contact detection can be integrated into dynamic analysis of human whole-body motions.

The contributions of this paper are summarized as follows:

- Application of novel learning method combining self-attention and VQ-VAE for contact detection and estimation of human motions
- Automatic contact detection as discrete "contact code" corresponding to different contact states with reduced efforts of manual annotation, together with reconstructed contact force
- Validation of the proposed method with the human gait database and demonstration of its utility through application to dynamic motion analysis

In this paper, following related work in Section II, the methodology for learning-based contact estimation and dynamic analysis is described in Section III. Results of contact estimation and detection with walking motions are presented together with dynamic analysis to validate the effectiveness of the proposed method in Section IV before concluding the paper.

II. Related Work

As mentioned earlier, contact detection from motion data or images has made considerable advances recently. Among them, BEHAVE [3] has been proposed as a dataset including whole-body interaction with objects including contacts tracked based on learning with multi-view RGB-D images. Huang et al. [2] proposed RICH dataset composed of human pose and shape scenes with more precise vertex-level 3D human-object contacts by resolving occlusions based on inference by applying Transformer mechanism. In those studies, contacts are estimated through the generalized human shape model SMPL (Skinned Multi-Person Linear model) [10], computed from image and geometry, by judging contact occurrence when the distance between human and objects are below $2.5\,\mathrm{cm}$. Other learning-based approaches to detect and estimate contacts directly from motion data have been proposed using video images [11], [12], inertial motion capture data [13], [14], [15] basically using machine learning techniques. It is nevertheless still challenging to differentiate multiple contacts at one body segment such as "toe-touchdown" and "heel-liftoff" during walking due to lack of precise association of motion with contacts. This makes inverse dynamics analysis difficult to estimate torques at the toe and ankle. While Mourot et al. [16] offers a dataset with insole sensor information to deal with distinct toe and heel contacts, it focuses estimation with the sensor force information and removal of its side effects. Kumano et al. [17] proposed a method for estimating human walking motion from inertial measuring units (IMUs) equipped at wrists and heels based on deep learning, but contact detection and estimation are out of their scope.

From the robotics perspective, Ramadoss et al. reported motion estimation of a human's floating base by combining dynamic optimization and inverse kinematics using motion sensors and shoes embedding force-torque sensors. Contacts are detected from the center of pressure measured with the sensorized shoes through extended Kalman Filter (EKF) [18]. Sugawara et al. proposed unsupervised motion segmentation based on learning from demonstration with a 6-axis force torque sensor signal for such tasks as peg-in-hole or bottle-lid opening [19]. Those studies deal with motion understanding from contacts while this paper seeks the other way around.

Understanding human behavior involves several distinct tasks, including action recognition and segmentation. In supervised learning approaches, action labels are predetermined, and the system matches input data to corresponding actions. For instance, methods proposed in [20], [21], [22] utilize supervised learning directly from video footage to detect actions. Action segmentation involves dividing a temporal sequence of input data into distinct actions. Several methods, such as those proposed in [23], [24], [25] utilize supervised learning to identify multiple action types and their boundaries within analyzed video frames. Those supervised methods, reliant on annotated motion data for action recognition, extract motion features effectively. However, they depend on human-mediated segmentation, requiring motion data of appropriate granularity. To address this limitation, we have proposed a method [26] that extracts feature representations for continuous temporal chunks of motion without preexisting human knowledge, leveraging a variational autoencoder (VAE) to generate a discrete latent space, such as a vector quantized-variational autoencoder (VQ-VAE) [7], [8] or a dynamical variational autoencoder (dVAE)[27]. Temporal relationships between frames are discerned using self-attention, as proposed in the Transformer architecture [6]. The discrete representation excels in identifying discontinuous points in motion data, aiding in action detection within non-segmented motion data.

III. Learning-based Contact Estimation and Dynamic Analysis

A. Detecting Contact Codes by Self-supervised Learning

The problem tackled in this paper is detecting the contact from motion data under the condition that most of the

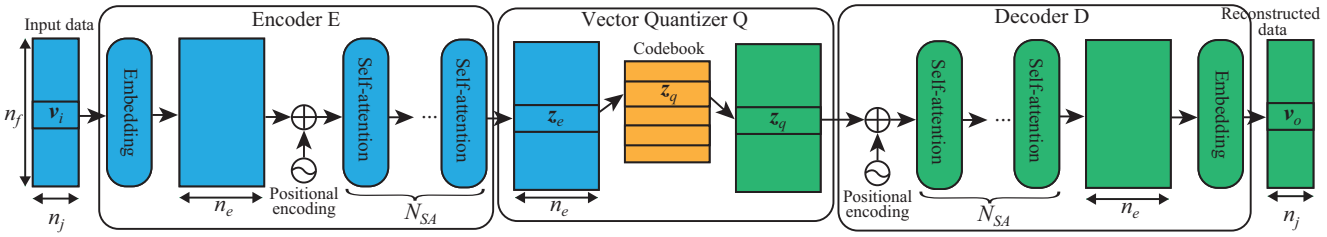

Fig. 2. The network converts the input motion vector $\boldsymbol{v}_i$ to the output force vector $\boldsymbol{v}_o$. The discrete latent vector $\boldsymbol{z}_q$ is extracted as the motion code to describe the state of contact.

data are without contact annotation. Therefore, the proposed method is based on the self-supervised approach proposed in [26] that generates a discrete latent space that describes the structure of a human motion. Since we try to model motions dominated by contact with the environment, the latent space is expected to express the contact information. The approach is based on an autoencoder combined with a block of clustering latent vectors. The clustering is implemented as vector quantization based on VQ-VAE [7], [8]. Since the discrete latent vectors are chosen from the codebook, we call them contact codes in this paper.

The architecture of the model of learning network is shown in Fig. 2. The input data of a motion consists of n_f frames of n_j-dimensional vectors at each frame, and the encoder converts the input vector $\boldsymbol{v}_i$ at each frame to a feature vector $\boldsymbol{z}_e$. The vector clustering replaces $\boldsymbol{z}_e$ with the vector $\boldsymbol{z}_q$, which is the nearest neighbor in the codebook based on Euclidean distance in the latent space. The decoder reconstructs the outputs $\boldsymbol{v}_o$ using $\boldsymbol{z}_q$. The input vector $\boldsymbol{v}_i$ at each frame is given from the motion capture as the joint angles. The output vector $\boldsymbol{v}_o$ is expected to express the force at the contact point produced by the motion and measured by force sensors. Therefore, the model is not an autoencoder, but a conversion from motion to force. The encoder and decoder are realized by self-attention layers [6]. The network is trained by minimizing the following loss L:

$$L = \sum_{k}^{n_f} \alpha L_{\text{reconst}}^{(k)} + L_{\text{latent}}^{(k)} \tag{1}$$

where $L_{\text{reconst}}^{(k)}$ and $L_{\text{latent}}^{(k)}$ are the losses of reconstruction and latent space, respectively, for the k-th frame of a motion sequence of n_f frames, and α is a user-defined weight of the reconstruction loss. The reconstruction loss is calculated as the difference between the output vector $\boldsymbol{v}_o$ and the measured force vector $\boldsymbol{f}$ as follows:

$$L_{\text{reconst}}^{(k)} = \| \boldsymbol{v}_{ok} - \boldsymbol{f}_k \|^2 \tag{2}$$

The latent space loss based on the vector quantization [7] is defined as follows:

$$L_{\text{latent}}^{(k)} = \| \text{sg}[\boldsymbol{z}_{qk}] - \boldsymbol{z}_{ek} \|^2 + \beta \| \boldsymbol{z}_{qk} - \text{sg}[\boldsymbol{z}_{ek}] \|^2 \tag{3}$$

where $\boldsymbol{z}_{ek}$ is the encoded vector at frame k and $\boldsymbol{z}_{qk}$ is a contact code in the codebook. The function sg is the stop-gradient operator that is defined as an identity at the forward computation time and has zero derivatives.

B. Relating Contact State and Learned Contact Code

This study aims to estimate the contact forces and states for various body segments based on human motion data. After learning the model presented in the previous subsection, the time-series data regarding contact forces and contact codes can be estimated from the human motion data through the model in the trained network. Although the contact code is expected to reflect the contact state of body segments, it is necessary to establish a correspondence between the contact code and the state. This subsection presents the method for constructing a mapping between the contact codes and states, by utilizing a limited dataset previously manually annotated for the contact states of body segments.

Let $q\ (\in \mathbb{N}) \leq n_q$ represent the index of the contact code, with the total number of the codes denoted as n_q. Let us assume that each code possesses a surjective mapping to the contact state of body segment B_j, whereas the mapping is not necessarily injective such that:

$$x_{j,q} = \begin{cases} 1 & (B_j \text{ contacts}) \\ 0 & (B_j \text{ does not contact}) \end{cases} \tag{4}$$

where $x_{j,q} \in \mathbb{B}\ (\triangleq \{0,1\})$ indicates the contact state of body segment B_j associated with contact code q. Namely, $x_{j,q}$ relates the learned contact codes to existence of contact at B_j, to be determined through optimization with annotated data.

The time-series of contact codes can be estimated from the human motion data using the model after learning. Let q_k denote the index of the estimated code at time sample k. The annotated contact state of body segment B_j at time sample k, denoted as $y_{j,k}$, can be formulated as:

$$y_{j,k} = \sum_{q=1}^{n_q} \delta_{q,q_k} x_{j,q} \tag{5}$$

where δ_{q,q_k} represents the Kronecker delta. The aforementioned equation can be concatenated and summarized in the following matrix form:

$$\boldsymbol{y}_j = \boldsymbol{S}_j \boldsymbol{x}_j \tag{6}$$

where $\boldsymbol{S}_j$ is the selection matrix, with each element value being zero or one.

Given the unknown mapping $\boldsymbol{x}_j$ (i.e. $x_{j,q}$ in Eq.(4)), the contact state $y_{j,k}$ can be estimated using Eq.(6). The unknown vector $\boldsymbol{x}_j$ will be determined by a limited dataset

containing annotated contact states. Let us solve the following optimization problem to obtain the unknown vector $\boldsymbol{x}_j$:

$$\min_{\boldsymbol{x}_j \in \mathbb{B}^{n_q}} \sum_{i}^{D_j} ||\widehat{\boldsymbol{y}}_j^{(i)} - \boldsymbol{S}_j^{(i)} \boldsymbol{x}||^2 \tag{7}$$

where, $\widehat{\boldsymbol{y}}_j^{(i)}$ represents the vector of annotated contact states of B_j obtained from data i, and $\boldsymbol{S}_j^{(i)}$ indicates the matrix associated with the codes estimated from data i, with the total number of manually annotated dataset denoted as D_j. While the problem in Eq.(7) constitutes a binary least squares problem, it can be formulated as an integer linear programming problem, as shown in Appendix I. The solver for an integer linear programming problem can be easily available, such as within MATLAB's optimization toolbox.

After obtaining $\boldsymbol{x}_j$ by solving Eq.(7) for all candidates of body segments (i.e. $\forall B_j$), the contact state of an arbitrary body segment at time samples k can be estimated from Eq.(5) by utilizing the estimated contact code q_k.

C. Dynamic Analysis

The measured or estimated contact forces and states will be utilized for estimating joint torques during movement. The equations of motion of an anthropomorphic system can be written as follows [28]:

$$\begin{bmatrix} \boldsymbol{H}_O(\boldsymbol{\theta}) \\ \boldsymbol{H}_J(\boldsymbol{\theta}) \end{bmatrix} \ddot{\boldsymbol{\theta}} + \begin{bmatrix} \boldsymbol{c}_O(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \\ \boldsymbol{c}_J(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{\tau}_J \end{bmatrix} + \sum_{c=1}^{N_c} \begin{bmatrix} \boldsymbol{K}_{O,c}(\boldsymbol{\theta}) \\ \boldsymbol{K}_{J,c}(\boldsymbol{\theta}) \end{bmatrix} \boldsymbol{f}_c \tag{8}$$

where, $\boldsymbol{\theta}$ represents the vector of generalized coordinates including the free-floating base (typically, pelvis segment) and the joints, $\boldsymbol{H}_O$ and $\boldsymbol{H}_J$ denote the inertia matrices corresponding to the floating base and the joints, respectively, $\boldsymbol{c}_O$ and $\boldsymbol{c}_J$ mean the terms associated with centrifugal, Coriolis, and gravity forces, $\boldsymbol{\tau}_J$ indicates the vector of joint torques, N_c is the number of contacted segments, $\boldsymbol{f}_c$ signifies the vector of external forces exerted on the contacted segment c, $\boldsymbol{K}_{O,c}$ and $\boldsymbol{K}_{J,c}$ are the matrices that map the forces in the Cartesian space to those in the joint space.

In the inverse dynamic analysis for human movements, the time-series data of joint coordinates $\boldsymbol{\theta}$ and their derivatives are obtained, usually through the inverse kinematics computation [29] utilizing the motion capture data. Prior to estimating the joint torque $\boldsymbol{\tau}_J$ in Eq.(8), the contact forces $\boldsymbol{f}_c$ exerted on each contacted segment has to be known. Given that the generalized forces of the floating base are zero, let us reformulate its equations in a simplified manner as follows:

$$\boldsymbol{K}_O \boldsymbol{F} = \boldsymbol{\tau}_O \tag{9}$$

where,

$$\boldsymbol{K}_O \triangleq \begin{bmatrix} \boldsymbol{K}_{O,1} & \cdots & \boldsymbol{K}_{O,N_c} \end{bmatrix} \tag{10}$$

$$\boldsymbol{F} \triangleq \begin{bmatrix} \boldsymbol{f}_1{}^T & \cdots & \boldsymbol{f}_{N_c}{}^T \end{bmatrix}^T \tag{11}$$

$$\boldsymbol{\tau}_O \triangleq \boldsymbol{H}_O \ddot{\boldsymbol{\theta}} + \boldsymbol{c}_O \tag{12}$$

As Eq.(9) represents the equations of only the floating base, if the number of contact segments exceeds one, the contact forces cannot be determined uniquely only from the equations. In typical human motion analysis, the contact forces are partially measured, for instance, by force sensors or force plates. Let us derive the contact forces by solving the optimization problem [30]:

$$\begin{aligned} \min_{\boldsymbol{F}} \quad & ||\boldsymbol{K}_O \boldsymbol{F} - \boldsymbol{\tau}_O||^2 + k_S ||\boldsymbol{K}_S \boldsymbol{F} - \boldsymbol{F}_S||^2 + k_R ||\boldsymbol{F}||^2 \\ \text{subject to} \quad & \boldsymbol{C}\boldsymbol{F} \geq \boldsymbol{0} \end{aligned} \tag{13}$$

where, k_S and k_R denote the weighting factors for the terms related to sensor errors and regularization, respectively, $\boldsymbol{F}_S = [\boldsymbol{f}_{S,1}^T \ \cdots \ \boldsymbol{f}_{S,N_S}^T]^T$ represents the concatenated vectors of the forces, denoted as $\boldsymbol{f}_{S,i}$, measured by the N_S set of sensor, the matrix $\boldsymbol{K}_S$ maps the forces in the sensor coordinates to those in the body segment coordinates, and $\boldsymbol{C}$ is the matrix derived from the physically consistent conditions regarding the reaction forces. It should be noted that the size of vectors F_S and F_C are not always equal, for instance, in cases when more than one body segment contact one sensor.

As the number of contacted segments (i.e. N_c) varies during motion, the contacted segments need to be known at each time instance in order to solve Eq.(13). In standard motion analysis, after manually annotating the contact states, the contact forces are estimated by solving Eq.(13) with the measured forces. Subsequently, the joint torques are obtained by utilizing Eq.(8). In the proposed framework, instead of using the manually annotated contact states and measured forces, the contact forces estimated by the learning model as well as the contact state estimation, described in the previous subsections, are utilized to solve Eq.(13).

We have validated the proposed method by applying it to walking motion data with ground-truth of contact force. The motions were taken from "AIST Gait Database" [5] that consists of walking motions of 300 subjects as shown in Fig. 3, each of them includes the body motion and ground reaction force (GRF), measured by a motion capture system and force plates respectively. We investigate the capacity of contact estimation from the motions, followed by dynamic analysis based on the detected contacts.

Fig. 3. An example of walking motion in AIST Gait Database [5].

IV. Results of Contact Learning and Dynamic Analysis

A. Learning Contacts

We adopt DhaibaWorks digital human model [9] that includes kinematic and dynamic models to represent the measured human walking motion and to apply dynamic analysis. The input vector $\boldsymbol{v}_i$ is motion representation based on DhaibaWorks with $n_j = 58$ converted from captured human motion data using inverse kinematics computation, composed of the three components of axis angle of 19 joints and the height of the root, which is the base position of the human body. The number of frames n_f is adjusted to 1000 through interpolation. The output $\boldsymbol{v}_o$ for each foot has a vector of 8 components composed of the following elements:

- 3D vector of ground truth of GRF $\boldsymbol{F}$ normalized by the subject's mass and reoriented with respect to the coordinate frame attached to each foot
- 3D vector of foot velocity $\boldsymbol{V}_{foot}$
- binary value B_F computed from of z-component of GRF as follows

$$B_F = \begin{cases} 1 & F_z > F_{thresh} \\ 0 & \text{otherwise} \end{cases}$$

- binary value $1 - B_F$

The idea behind using both foot force and velocity is exploiting their complementary property to help the network learn the switching timing of touchdown and liftoff at the toe and heel. The force value is nonzero while the velocity is zero during the stance phase of a foot, and vice versa during the swing phase. The binary values are introduced for better contact detection. Here the threshold F_{thresh} for the normalized vertical force is 0.02.

In the Gait Database, the whole stance phase of a foot occupies at most 700 steps out of 1000. We have then chosen 100 motion data from different subjects and sliced 700 frames of data starting at a random frame between 1 and 300. The proposed VQ-VAE network was trained for each foot with 80 out of 100 motion data for 5000 iterations with the weight of reconstruction loss $\alpha = 100$ in Eq.(1), with the learning rate of 2.0×10^{-6}. The remaining 20 are used

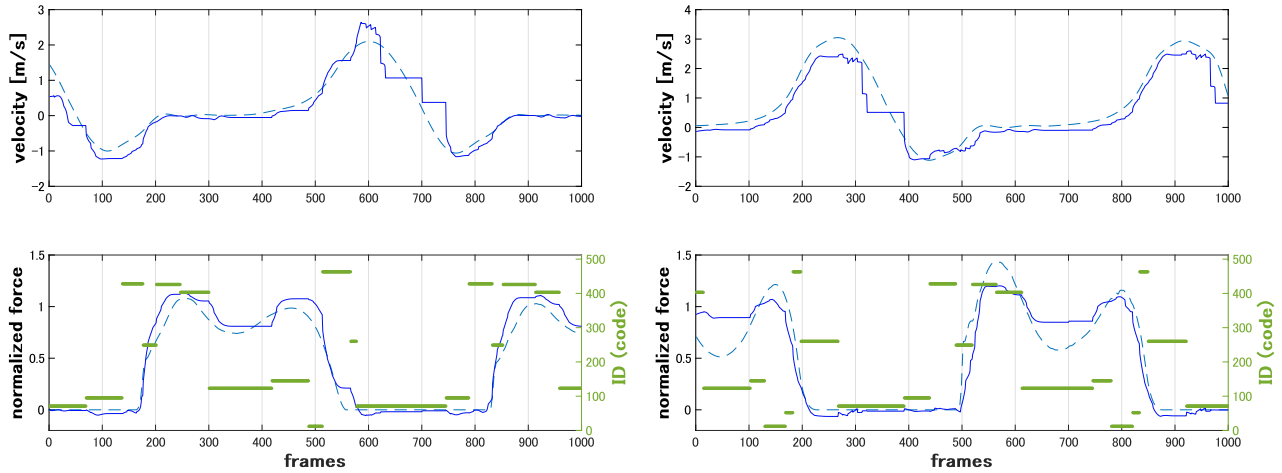

Fig. 4. Reconstructed vertical foot velocity m/s (upper row) / normalized force (lower row) at the local foot coordinate with contact codes for the right foot for motions starting with (A) swing phase and (B) stance phase. These motions are from different subjects. The measured and reconstructed values are plotted with dotted light blue and solid blue lines with contact codes (IDs) in solid green line segments on the right-side axis.

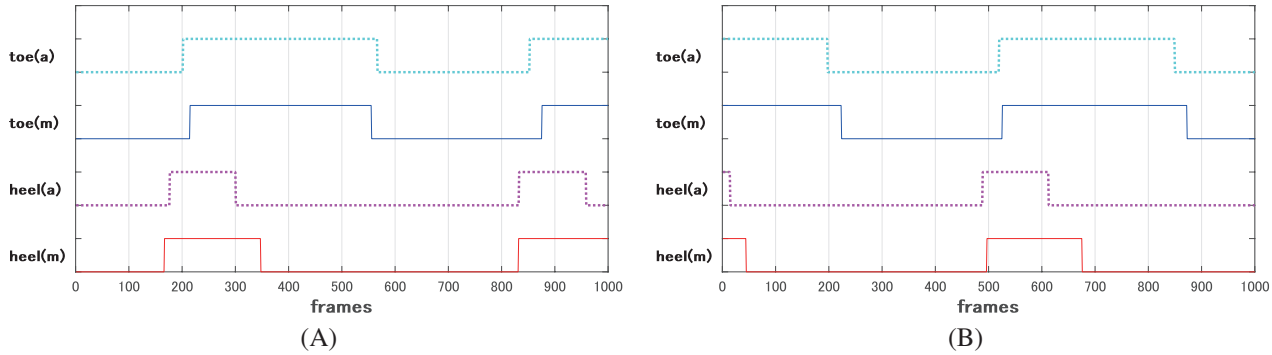

Fig. 5. Comparison of contact states estimated with the proposed method and manually annotated contact detection of the right foot for the same motions presented in Fig. 4. The estimated contact states (1: contact, 0: no contact) at the toe (dotted light and solid dark blue color) and heel (dotted cyan and solid red colors) are shown (a) and (m) denoting automatic and manual annotation respectively.

as test data for validation. The software was implemented on pyTorch 2.20 and run with batch size 8 on an NVIDIA RTX A6000 48GB Memory and the training took around 17 hours for each foot. Since we set larger weight on reconstruction, its loss $\sum_k^{n_f} L_{\text{reconst}}^{(k)}$ in Eq.(2), square sum of difference of normalized force, dropped from 10.71 and 12.33 to 0.0646 and 0.0488 for left and right foot respectively after the training.

Total 35 and 25 contact codes have been extracted for the left and right foot through the training. Figure 4 shows two examples from different subjects of the original (dotted light blue lines) and reconstructed (dark blue lines) normalized vertical contact force and velocity $\mathrm{m/s}$ at the local coordinate frame fixed at the right foot, out of 20 test data that were not used for training. The solid green lines are the contact codes $\boldsymbol{z}_q$ (ID in the graph) obtained from the input motion. Those code ID numbers at the right-side axis are attributed arbitrarily during the training and their value does not have meaning. As can be seen, the network reproduces the two-peak profile of the force and corresponding velocity with sufficient precision even with the data starting in the middle of the stance phase. The root mean square error (RMSE) of the force normalized with subjects' weight at each step over all 20 test validation data was 19.9% and 13.5% for left and right foot.

The same patterns of code sequence are observed where the contact forces change sharply in different data, not only around the addition/removal of contacts but also at the changing point of force increase/decrease during contact. The latter corresponds to the toe touchdown and heel liftoff that is difficult to detect only from the force information, which would therefore require manual annotation from motion capture data.

B. Contact state estimation from contact codes

One contact code is selected at each frame as shown in Fig. 2. Assuming that the frequently used codes appear as the main descriptors covering the majority of contact motions, we can expect to reduce manual annotation tasks by using a limited number of motions covering those representative contact codes. We filtered out rarely appearing codes by applying the limit of minimum n_{min} appearance of at least during total n_f frames and T_{min} times out of 80 training data. With $n_{min} = 3$ and $T_{min} = 5$, 17 and 16 contact codes out of 35 and 25, respectively for left and right foot, remain as frequently used codes.

We have chosen $D_j = 19$ motions from 80 training data as $\widehat{\boldsymbol{y}}_j^{(i)}$ in Eq.(7) to cover the filtered codes and manually annotated them to differentiate the toe and heel contacts. We then applied the method in III-B to the 20 test data. In order to validate the effectiveness of the proposed automatic annotation, we also manually annotated all the test data and compared them with the automatically detected contacts shown as binary values denoting the existence of contact in Fig. 5 for the same motions as Fig. 4.

The figures show that the estimated contact states with the proposed method match the annotated ones well, including "toe-touchdown" and "heel-liftoff" during the support phase which was difficult with existing methods, based on less than 25% of manual annotation of training data. The average value of the sum of square error between manually and automatically annotated contact information (the cost function in Eq.(7)) per motion sequence of 1000 frames over 20 test data, representing the accuracy of automatic contact detection, was 59.9 and 103.1 for the toe and heel respectively.

C. Application to dynamics analysis

The results of the previous subsections are applied to dynamic analysis presented in III-C for the motions shown in Figs. 4 and 5 through implementation on MATLAB. The contact forces $\boldsymbol{F}$ in Eq.(13) and then joint torques $\boldsymbol{\tau}_J$ in Eq.(8) at the leg are computed by using the estimated forces in IV-A based the proposed learning method with contact states obtained in IV-B. The results of learning-based dynamic analysis are shown in Figs. 6 and 7 with blue lines, compared with the contact force computed with measured force and manually annotated contact states with dotted red lines. Note that the forces and torques are with respect to the world coordinate frame with the elapsed time in seconds instead of frames. In Fig. 6, we can see that learning-based contact forces are in good accordance with the measured ones in terms of magnitude and profile. It can also be observed that the time discrepancy of contact state transition in Fig. 5 causes a large error of contact removal at the ankle. Figure 7 plots the estimated torque around the horizontal axes vertical to the sagittal plane for the toe, ankle, and knee joints during walking. Precise analysis of those joints became possible by detecting distinct contacts of the toe and heel while the measured force usually comes from the same force plate. The general tendency of the torque from learning-based force

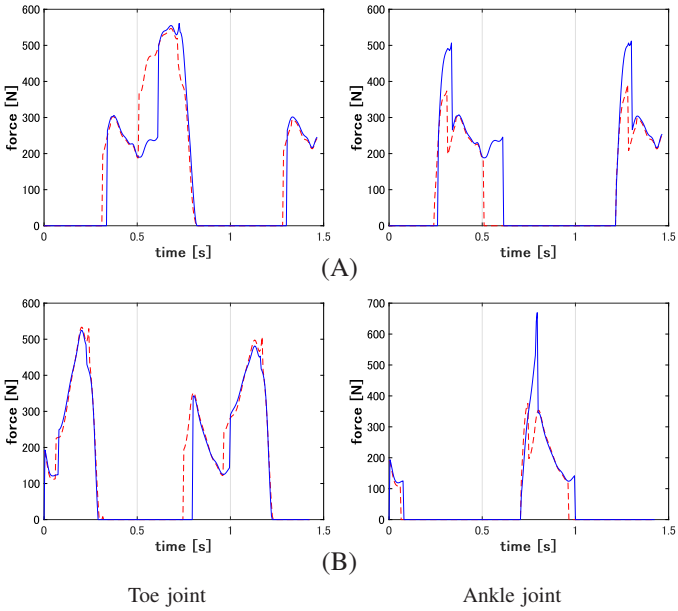

Fig. 6. Comparison of contact forces computed using estimated force and automatically annotated contact state (blue) and measured force and manually annotated contact state (red) at the toe and ankle of the right foot. The motions are from Fig. 4 starting (A) stance phase and (B) swing phase.

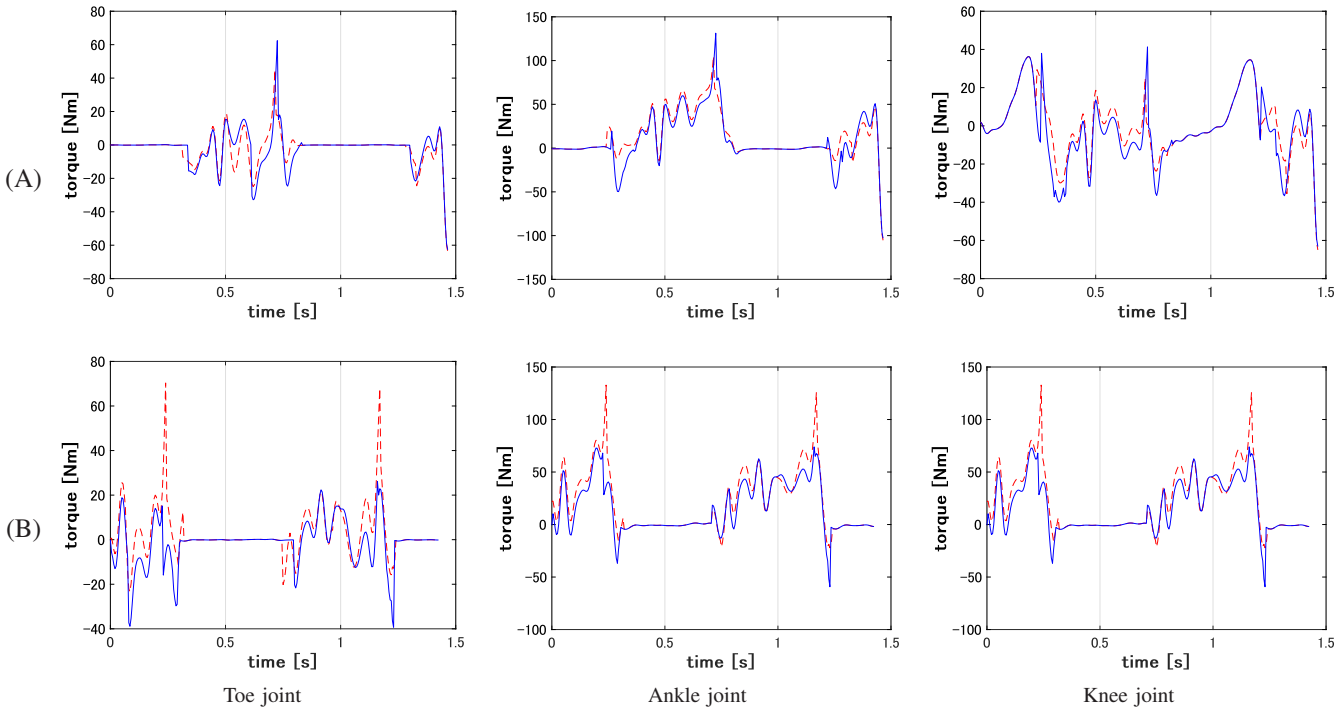

Fig. 7. Comparison of joint torque computed by estimated force and automatically annotated contact state (blue) and measured force and manually annotated contact state (dotted red) at the toe, ankle, and knee joint of the right foot. The motions are from Fig. 4 starting (A) stance phase and (B) swing phase.

(blue) estimation corresponds well to the estimated torque with measured force and manual annotation (dotted red). The torques from learning-based estimation are rather jerky compared to the values based on the measurement. One reason is the lack of smoothness of The estimated forces in Fig. 4. This issue can be addressed by applying a smoothing function, possibly trained together with the motion.

Overall, although there is still room for improvement, we believe the first trial of learning-based contact estimation from motion brought promising results for dynamic analysis. We expect to apply this framework not only to the investigation of large-scale and long-term walking gait datasets without force information such as AMASS [31], but also to various contact-rich human motion analyses like workload evaluation during carrying tasks or regulating the supportive contact force of assistive robot, by enhancing this framework for other contact motions.

V. Conclusions

This paper presented a method for contact detection and force estimation from human motion to solve the critical problem of labor-intensive annotation of contacts and also in view of providing the motion-only datasets with contact information. A learning framework based on self-attention and VQ-VAE is proposed to extract the discrete feature values called "contact codes" that correspond to different contact states and allow the reconstruction of contact force information. The network can then convert the input motion into a sequence of codes, which can be decoded to contact states and forces. After training the network with motion inputs and force outputs, optimization has been introduced to relate the small number of manually annotated contacts with the obtained contact codes. We have applied the proposed method to the human gait database with ground-truth force data. A reduced number of manually annotated training data resulted in sufficiently accurate foot contact detection, especially the transition from toe touchdown to heel liftoff during the stance phase, as well as force estimation of the validation test data. This demonstrates the effectiveness of the method towards minimal manual annotation for dynamic motion analysis. It is noteworthy that contact forces and states can be estimated accurately enough from motion data, paving a new possibility of dynamic motion analysis using the motion dataset without force information.

Future directions include extension of the method towards other whole-body motions than the current single dataset of walking and foot contacts. This will need investigation of modeling multiple contacts between the surface body shape model like Dhaiba and SMPL and the environments. It is also important to collect data on whole-body motions involving contacts together with force/tactile sensor information as a reference for learning. We will tackle those challenges of analysis and data preparation as the next step of this research.

Appendix I
Conversion from Binary Least Squares to Integer Linear Programming

Let us consider the optimization problem formulated as the following quadratic form with respect to $\boldsymbol{x} \in \mathbb{B}^n$:

$$\min_{\boldsymbol{x}\in\mathbb{B}^n} \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{d}^T\boldsymbol{x} \tag{14}$$

The components, $\boldsymbol{Q} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{d} \in \mathbb{R}^n$, are obtained from the binary least squares problem denoted as Eq.(7).

To address the aforementioned problem, let us introduce a vector $\boldsymbol{z} \in \mathbb{B}^{n^2+n}$ whose elements correspond to the first- or second-order polynomials of each element of $\boldsymbol{x}$, denoted as $x_i \in \mathbb{B}$ $(1 \leq i \leq n)$, as follows:

$$\boldsymbol{z} \triangleq \begin{bmatrix} x_1^2 & x_1x_2 \cdots & x_1x_n & x_2x_1 \cdots & x_{n-1}x_n & x_n^2 & \boldsymbol{x}^T \end{bmatrix}^T \tag{15}$$

As x_i takes a binary value, the following relationships hold between the first- and second-order polynomials:

$$x_i x_j \leq 1 - x_i - x_j \tag{16}$$

$$x_i x_j \leq x_i \tag{17}$$

$$x_i x_j \leq x_j \tag{18}$$

The above three equations can be summarized as the following linear form with matrix $\boldsymbol{A} \in \mathbb{B}^{3n^2 \times 3n^2}$ and vector $\boldsymbol{b} \in \mathbb{B}^{3n^2}$:

$$\boldsymbol{A}\boldsymbol{z} \leq \boldsymbol{b} \tag{19}$$

Finally, the problem in Eq.(14) can be formulated as an integer linear programming problem by utilizing the vector $\boldsymbol{z}$ and the constraint in Eq.(19) as followings:

$$\begin{aligned} &\min_{\boldsymbol{z} \in \mathbb{Z}^K} \quad \boldsymbol{c}^T \boldsymbol{z} \\ &\text{subject to} \quad \widehat{\boldsymbol{A}}\boldsymbol{z} \leq \widehat{\boldsymbol{b}}, \quad \boldsymbol{0} \leq \boldsymbol{z} \leq \boldsymbol{1} \end{aligned} \tag{20}$$

where, $\boldsymbol{c} \in \mathbb{R}^{n^2+n}$ is comprised of the vector $\boldsymbol{d}$ and the elements $Q_{i,j}$ of the matrix $\boldsymbol{Q}$ such that:

$$\boldsymbol{c} \triangleq \begin{bmatrix} Q_{1,1} & \cdots & Q_{1,n} & \cdots & Q_{n-1,n} & Q_{n,n} & \boldsymbol{d}^T \end{bmatrix}^T \tag{21}$$

REFERENCES

[1] E. Yoshida, "Towards understanding and synthesis of contact-rich anthropomorphic motions through interactive cyber-physical human," *Frontiers in Robotics and AI*, vol. 9, 2022.

[2] C.-H. P. Huang *et al.*, "Capturing and inferring dense full-body human-scene contact," in *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, 13 274–13 285.

[3] B. L. Bhatnagar *et al.*, "BEHAVE: Dataset and method for tracking human object interactions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2022, 15 935–15 946.

[4] C. Mandery *et al.*, "Unifying representations and large-scale whole-body motion databases for studying human motion," *IEEE Transactions on Robotics*, vol. 32, no. 4, 796–809, 2016.

[5] Y. Kobayashi *et al.*, "AIST gait database 2019," https://unit.aist.go.jp/harc/ExPART/GDB2019.html, 2019.

[6] A. Vaswani *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon *et al.*, Eds., vol. 30. Curran Associates, Inc., 2017, 5998–6008.

[7] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," *CoRR*, vol. abs/1711.00937, 2017. [Online]. Available: http://arxiv.org/abs/1711.00937

[8] A. Razavi, A. van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with VQ-VAE-2," in *Advances in Neural Information Processing Systems*, H. Wallach *et al.*, Eds., vol. 32, 2019.

[9] Y. Endo, T. Maruyama, and M. Tada, "Dhaibaworks: A software platform for human-centered cyber-physical systems," *International Journal of Automation Technology*, vol. 17, no. 3, 292–304, 2023.

[10] M. Loper *et al.*, "SMPL: A skinned multi-person linear model," *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, vol. 34, no. 6, 248:1–248:16, Oct. 2015.

[11] Z. Li *et al.*, "Estimating 3d motion and forces of person-object interactions from monocular video," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, 8632–8641.

[12] N. Louis *et al.*, "Learning to estimate external forces of human motion in video," in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 3540–3548. [Online]. Available: https://doi.org/10.1145/3503161.3548377

[13] A. Karatsidis *et al.*, "Estimation of ground reaction forces and moments during gait using only inertial motion capture," *Sensors (Basel)*, vol. 17, no. 1, p. 75, 2016.

[14] T.-H. Pham, S. Caron, and A. Kheddar, "Multicontact interaction force sensing from whole-body motion capture," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, 2343–2352, 2018.

[15] H. Ma *et al.*, "Real-time foot-ground contact detection for inertial motion capture based on an adaptive weighted naive bayes model," *IEEE Access*, vol. 7, 130 312–130 326, 2019.

[16] L. Mourot *et al.*, "Underpressure: Deep learning for foot contact detection, ground reaction force estimation and footskate cleanup," *Computer Graphics Forum*, vol. 41, no. 8, 195–206, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.14635

[17] Y. Kumano *et al.*, "Estimating whole-body walking motion from inertial measurement units at wrist and heels using deep learning," *International Journal of Automation Technology*, vol. 17, no. 3, 217–225, 2023.

[18] P. Ramadoss *et al.*, "Estimation of human base kinematics using dynamical inverse kinematics and contact-aided lie group kalman filter," in *Proc. IEEE-RAS Int. Conf. on Humanoid Robots*, 2022, 364–369.

[19] K. Sugawara, S. Sakaino, and T. Tsuji, "Unsupervised human motion segmentation based on characteristic force signals of contact events," *IEEE Robotics and Automation Letters*, vol. 8, no. 10, 6203–6210, 2023.

[20] R. Girdhar *et al.*, "Actionvlad: Learning spatio-temporal aggregation for action classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, 971–980.

[21] X. Wang *et al.*, "Oadtr: Online action detection with transformers," *arXiv preprint arXiv:2106.11149*, 2021.

[22] G. A. Sigurdsson *et al.*, "Asynchronous temporal fields for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, 585–594.

[23] Y. A. Farha and J. Gall, "Ms-tcn: Multi-stage temporal convolutional network for action segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, 3575–3584.

[24] Y. Huang, Y. Sugano, and Y. Sato, "Improving action segmentation via graph-based temporal reasoning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, 14 024–14 034.

[25] M.-H. Chen *et al.*, "Action segmentation with joint self-supervised temporal domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, 9454–9463.

[26] T. Abe *et al.*, "Self-supervised extraction of human motion structures via frame-wise discrete features," *arXiv e-prints*, p. arXiv:2309.05972, 2023.

[27] A. Ramesh *et al.*, "Zero-shot text-to-image generation," *CoRR*, vol. abs/2102.12092, 2021. [Online]. Available: https://arxiv.org/abs/2102.12092

[28] Y. Fujimoto, S. Obata, and A. Kawamura, "Robust biped walking with active interaction control between foot and ground," in *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, 1998, 2030–2035.

[29] K. Ayusawa *et al.*, "Fast inverse kinematics based on pseudo-forward dynamics computation: Application to musculoskeletal inverse kinematics," *IEEE Robotics and Automation Letters*, vol. 8, no. 9, 5775–5782, 2023.

[30] K. Yamane, Y. Fujita, and Y. Nakamura, "Estimation of physically and physiologically valid somatosensory information," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, 2624–2630.

[31] N. Mahmood *et al.*, "AMASS: Archive of motion capture as surface shapes," in *International Conference on Computer Vision*, Oct. 2019, 5442–5451.